Linux 系统编程-基本命令

传智播客-邢文鹏[1]

2014-09-09

[1]http://blog.csdn.net/itcastcpp

前言

学习目标

成为linux/unix系统程序员

学习态度

* 谦虚
* 严谨
* 勤思
* 善问

学习方法

只听不练肯定学不会Linux，每个知识点都需要去动手实践

目录

第 1 章

基本命令

第 2 章

shell

2.1 shell家族

shell：命令解释器，根据输入的命令执行相应命令。

察看当前系统下有哪些shell：

```
cat /etc/shells
```

察看当前系统正在使用的shell

```
echo $SHELL
```

常见shell：

```
/bin/sh (已经被 /bin/bash 所取代)
/bin/bash (就是 Linux 默认的 shell)
/bin/ksh (Kornshell 由 AT&T Bell lab. 发展出来的, 兼容于 bash)
/bin/tcsh (整合 C Shell , 提供更多的功能)
/bin/csh (已经被 /bin/tcsh 所取代)
/bin/zsh (基于 ksh 发展出来的, 功能更强大的 shell)
```

2.2 bash

bash 是一个为GNU计划编写的Unix shell。它的名字是一系列缩写：Bourne-Again SHell — 这是关于Bourne shell（sh）的一个双关语（Bourne again / born again）

bash是许多Linux平台的内定Shell，事实上，还有许多传统UNIX上用的Shell，像tcsh、csh、ash、bsh、ksh等等，Shell Script大致都类同，当您学会一种Shell以后，其它的Shell会很快就上手，大多数的时候，一个Shell Script通常可以在很多种Shell上使用。

bash是大多数Linux系统以及Mac OS X v10.4默认的shell，它能运行于大多数Unix风格的操作系统之上，甚至被移植到了Microsoft Windows上的Cygwin系统中，以实现windows的POSIX虚拟接口。此外，它也被DJGPP项目移植到了MS-DOS上。

2.3 命令和路径补齐

在bash下敲命令时，Tab键可以补全已经敲了一部分的文件名和目录名。如果是Ubuntu系统，系统默认启用了bash completion，还可以补全命令的某些参数、Makefile目标等等。如果是Debian系统，可以用以下命令启用bash completion：

```
$ source /etc/bash_completion
```

建议将这一行加入~/.bashrc启动脚本中。比如使用sudo后面接命令，如果没有bash completion则只有sudo可以补全，后面的命令不能补全。如果启用了bash completion，则后面的命令，包括命令的某些参数（比如aptitude命令的install）都可以补全了。

比如在主目录下要列出桌面目录的内容，输入（不回车）

```
$ ls De
```

然后敲Tab键，如果以De开头的文件或文件夹只有Desktop一个，就自动补全为

```
$ ls Desktop
```

否则，再敲一次Tab键，将会把所有以De开头的文件或文件夹列在下面供你选择(在这里我们手动创建另外一个以De开头的文件)

```
$ touch Death
$ ls De
Death    Desktop/
```

你可以再补敲一个s再Tab，这次Desktop就会补全到命令后面了。

有的人是从DOS时代过来的，留下一个很不好的习惯就是在找一个文件时反复地cd、ls、cd、ls。。。等找到了要找的文件时再想回到先前的目录，已经不记得先前是从哪个目录转到这里来的了。

我们从上面可以看出，Tab补全本身就具备了ls的功能，上面的Tab补全相当于ls -Fd De*命令。所以我们完全不必反复地cd到别的目录然后ls去找文件，多按几次Tab就可以一条命令完成了，这样的好处是我们的当前目录不用变，不需要找完了文件再cd回来，同时省去了大量的按键次数。更重要的是，自动补全同时兼具了检查拼写错误的功能，如果前面几个字母拼写错了，就补全不出东西来，用户就知道拼写错了，如果前面几个字母没有拼写错，那么由系统补全出来的文件名肯定也不会有拼写错误，避免了用户在敲很长的文件名时易犯的拼写错误。

2.4 历史记录

历史记录是另外一个非常方便的功能。按上下移动光标键（或者Ctrl-p、Ctrl-n）可以一条一条浏览以前输过的命令。如果有需要重复输入的命令就不用输第二次了。如果你能记住以前输过的某条命令中的某个关键字，可以按Ctrl-r，然后输入关键字，随着你每输入一个字母，bash会做增量式（increasingly）查找，也可以反复按Ctrl-r或Ctrl-s向前向后查找。如果找到了，按左右移动光标键或Home键(Ctrl-a)或End键(Ctrl-e)将该命令带回提示符下进一步修改，或者直接按Enter键原封不动地执行该命令。

2.5 主键盘快捷键

bash的快捷键和emacs保持一致，用惯其中之一再用另一个程序会很顺手的。请记住一条原则：尽量使用主键盘快捷键而不使用移动光标键和编辑键。因为手不必离开主键盘是效率最高的，这样在你一生之中所节省的来回移动手的时间绝对可以用星期来计算，是绝对值得你花十分钟的时间记住这些快捷键的。

功能	快捷键	助记
上	Ctrl-p	previous
下	Ctrl-n	next
左	Ctrl-b	backward
右	Ctrl-f	forward
Del	Ctrl-d	delete光标后面的
Home	Ctrl-a	the first letter
End	Ctrl-e	end
Backspace	Backspace	delete光标前面的

第 3 章

目录和文件

3.1 类Unix系统目录结构

ubuntu没有盘符这个概念，只有一个根目录/，所有文件都在它下面

```
/    根目录
    bin       //系统可执行程序，如命令
    boot     //内核和启动程序，所有和启动相关的文件都保存在这里
         grub    //引导器相关文件
    dev      //设备文件
    etc      //系统软件的启动和配置文件，系统在启动过程中需要读取的文件都在这个目录。如LILO参数、用户账户和密码。
    home   //用户的主目录。下面是自己定义的用户名的文件夹
    lib       //系统程序库文件,这个目录里存放着系统最基本的动态链接共享库，类似于Windows下的system32目录，几乎所有的应用程序都需要用到这些共享库。
    media  //挂载媒体设备，如光驱、U盘等
    mnt    //目录是让用户临时挂载别的文件系统，如挂载Windows下的某个分区，ubuntu默认还是挂载在/media目录。
    opt     //可选的应用软件包（很少使用）
    proc     //这个目录是系统内存的映射，我们可以直接访问这个目录来获取系统信息。也就是说，这个目录的内容不在硬盘上而是在内存里。
    sbin    //管理员系统程序
    selinux
    srv
    sys   //udev用到的设备目录树，/sys反映你机器当前所接的设备
    tmp  //临时文件夹
    usr   //这是个最庞大的目录，我们要用到的很多应用程序和文件几乎都存放在这个目录下。]
          bin    // 应用程序
         game  //游戏程序
         include
         lib    //应用程序的库文件
         lib64
         local   //包含用户程序等
         sbin    //管理员应用程序
```

3.2 用户目录

位于/home/user，称之为用户工作目录或家目录,表示方式：

```
/home/user
~
```

3.2.1 相对路径和绝对路径

绝对路径

从/目录开始描述的路径为绝对路径，如：

```
cd /home
ls /usr
```

相对路径

从当前位置开始描述的路径为相对路径，如：

```
cd ../../
ls abc/def
```

.和..

每个目录下都有.和..

. 表示当前目录

.. 表示上一级目录，即父目录

根目录下的.和..都表示当前目录

3.3 ls

ls [OPTION]… [FILE]…

ls是英文单词list的简写，其功能为列出目录的内容。这是用户最常用的一个命令，因为用户需要不时地查看某个目录的内容。该命令类似于DOS下的dir命令。 对于每个目录，该命令将列出其中的所有子目录与文件。对于每个文件，ls将输出其文件名以及所要求的其他信息。默认情况下，输出条目按字母顺序排序。当未给出目录名或是文件名时，就显示当前目录的信息。

主要的OPTION有：

```
-a 列出隐藏文件，文件中以“.”开头的均为隐藏文件，如：~/.bashrc
-l 列出文件的详细信息
-R 连同子目录中的内容一起列出
```

用ls -l命令显示的信息中，开头是由10个字符构成的字符串，其中第一个字符表示文件类型，它可以是下述类型之一：

```
- 普通文件
d 目录
l 符号链接
b 块设备文件
c 字符设备文件
s socket文件，网络套接字
p 管道
```

后面的9个字符表示文件的访问权限，分为3组，每组3位。第一组表示文件属主的权限，第二组表示同组用户的权限，第三组表示其他用户的权限。每一组的三个字符分别表示对文件的读、写和执行权限。各权限如下所示：

```
r 读
w 写
x 可执行。对于目录，表示进入权限。
s 当文件被执行时，把该文件的UID或GID赋予执行进程的UID（用户ID）或GID（组 ID）。
t 设置标志位（sticky bit）。如果是有sticky bit的目录，在该目录下任何用户只要有适当的权限即可创建文件，但文件只能被超级用户、目录拥有者或文件属主删除。如果是有sticky bit的可执行文件，在该文件执行后，指向其正文段的指针仍留在内存。这样再次执行它时，系统就能更快地装入该文件。
- 没有相应位置的权限。
```

访问权限后面的数字表示与该文件共享inode的文件总数，即硬链接数(参见下面ln命令)。

3.4 cd

change dir 改变当前所在路径

```
cd ~
cd dir1/dir2
cd ..
```

3.5 which

查看指定命令所在路径

```
which ls
```

3.6 pwd

查看当前所在路径

```
pwd
```

3.7 mkdir

mkdir [OPTION] DIRECTORY…

创建目录DIRECTORY，可以一次创建多个。OPTION如果是-p，表示可以连同父目录一起创建。

3.8 rmdir

rmdir [OPTION]… DIRECTORY…

删除空目录，可以一次删除多个。OPTION如果是-p，表示可以连同空的父目录一起删除。mkdir和rmdir的用法举例：

```
$ mkdir a
$ mkdir a/b
$ ls a
b
$ rmdir a/b
$ ls a
$ rmdir a
$ mkdir a/b
mkdir: cannot create directory `a/b': No such file or directory
$ mkdir -p a/b
$ rmdir -p a/b
```

3.9 touch

touch [OPTION]… FILE…

* 将每个文件的访问及修改时间都更新为目前的时间。
* 如果文件不存在，则创建一个字节数为0的文件。

3.10 rm

删除文件：

```
rm file
```

删除目录：

```
rm dir -rf
```

3.11 mv

重命名：

```
mv file1 file2
```

移动文件:

```
mv file1 ~/
```

3.12 cp

拷贝文件：

```
cp file1 file2
cp file1 dir/
cp file1 ../
```

拷贝目录:

```
cp dir1 dir2 -r
cp dir1 ~/ -r
```

3.13 cat

查看文件里内容，输出到终端，如果cat时没跟文件名，则读标准输入，遇到\n后，输出到标准输出，终端下输入Ctrl-d表示结束

3.14 more

more [OPTION] FILE…

查看文本文件的内容，屏幕显示完一屏就等待用户按下任意键再滚动到下一屏，如果中途不想继续看下去了，可以按Ctrl+C或q终止显示。

3.15 less

less [OPTION] FILE…

查看文本文件的内容，屏幕显示完一屏就等待用户按键，用户可以向上或向下查看，如果中途不想继续看下去了，可以按Ctrl+C或q终止显示。

3.16 head

head [OPTION]… FILE…

显示指定文件的前面几行。如果没有指定文件，将从标准输入（键盘）上读取。如果没有指定要显示的行数，则默认显示前10行。如果要显示文件的前5行：

```
$ head -5 file1
```

3.17 tail

tail [OPTION]… FILE…

显示文件的最后几行。若没有指定显示的行或字符数，则默认显示末尾10行。如果要显示文件末5行：

```
$ tail -5 file1
```

3.18 ln

链接有两种，一种被称为硬链接（Hard Link），另一种被称为符号链接（Symbolic Link）。建立硬链接时，链接文件和被链接文件必须位于同一个文件系统中，并且不能建立指向目录的硬链接。而对符号链接，则不存在这个问题。默认情况下，ln产生硬链接。如果给ln命令加上-s选项，则建立符号链接。举例如下，注意ls -l列出文件的硬链接数和字节数：

硬链接：

```
touch hello
ln  hello word_h
```

软链接：

```
ln -s hello word_s
```

3.19 tree

这个命令需要下载安装，ubuntu下

```
sudo apt-get install tree
```

按结构树的形状显示目录和文件

3.20 wc

利用 wc 指令我们可以计算文件的 Byte 数、字数、或是列数,若不指定文件名称、或是所给予的文件名为“-”,则 wc 指令会从标准输入设备读取数据。

```
wc -l ./*
```

-c 或—bytes 或—chars 只显示 Bytes 数。
-l 或—lines 只显示列数。
-w 或—words 只显示字数。

3.21 od

```
od -tcx file1
```

-t 指定数据的显示格式，主要的参数有：

```
c ASCII字符或反斜杠序列

d[SIZE] 有符号十进制数,每个整数SIZE字节。

f[SIZE] 浮点数,每个整数SIZE字节。

o[SIZE] 八进制（系统默认值为02）,每个整数SIZE字节。

u[SIZE] 无符号十进制数,每个整数SIZE字节。

x[SIZE] 十六进制数,每个整数SIZE字节。
```

3.22 du

查看某个目录的大小：
以M为单位

```
du -hm /home/xingwenpeng/test
```

以B为单位

```
du -hb ./*
```

以K为单位,4k的整数倍

```
du -hk ./*
```

3.23 df

df查看磁盘使用情况

```
df --block-size=GB
df --block-size=MB
```

第 4 章

文件属性和用户用户组

4.1 whoami

查看当前登陆用户

4.2 chmod

* 文字设定法

```
chmod [who] [+|-|=] [mode] 文件名
```

操作对象who可是下述字母中的任一个或者它们的组合：

```
u 表示“用户（user）”，即文件或目录的所有者。
g 表示“同组（group）用户”，即与文件属主有相同组ID的所有用户。
o 表示“其他（others）用户”。
a 表示“所有（all）用户”。它是系统默认值。
```

操作符号可以是：

```
+ 添加某个权限。
- 取消某个权限。
= 赋予给定权限并取消其他所有权限（如果有的话）。
```

设置mode所表示的权限可用下述字母的任意组合：

```
r 可读。
w 可写。
x 可执行。
```

* 数字设定法

chmod [mode] 文件名

我们必须首先了解用数字表示的属性的含义：

```
0表示没有权限，
1表示可执行权限，
2表示可写权限，
4表示可读权限，
```

然后将其相加。所以数字属性的格式应为3个从0到7的八进制数，其顺序是（u）（g）（o）。

例如，如果想让某个文件的属主有“读/写”二种权限，需要把4（可读）+2（可写）＝6（读/写）。

比如设置一个文件允许所有用户可写

```
$ chmod a+w file1
```

设置一个文件允许所有用户可读、可写、不可执行

```
$ chmod 666 file1

    user       group      other
  r   w   x   r  w  x   r   w   x
  4   2   1   4  2  1   4   2   1
     5          6          3
```

4.3 chown

chown [OPTION]… [OWNER:GROUP] FILE…

chown [OPTION]… —reference=RFILE FILE…

更改某个文件或目录的属主和属组。这个命令也很常用。例如root用户把自己的一个文件拷贝给用户A，为了让用户A能够存取这个文件，root用户应该把这个文件的属主设为A，否则，用户A无法存取这个文件。

OPTION的主要参数：

* -R 递归式地改变指定目录及其下的所有子目录和文件的拥有者。
* -v 显示chown命令所做的工作。

比如把一个文件改为itcast用户和nogroup用户组所有

```
$ sudo chown itcast:nogroup file1
```

注意：

* chown需要特权用户才能执行
* 一个文件的owner和owning group是没有关联的。一个文件属于用户A，也属于用户组B，并不表示用户A属于用户组B。

4.4 chgrp

chgrp [OPTION]… GROUP FILE…

chgrp [OPTION]… —reference=RFILE FILE…

该命令改变（指定）指定文件所属的用户组。其中group可以是用户组ID，也可以是/etc/group文件中用户组的组名。文件名是以空格分开的要改变属组的文件列表，支持通配符。如果用户不是该文件的属主或超级用户，则不能改变该文件的组。

OPTION的主要参数：

-R 递归式地改变指定目录及其下的所有子目录和文件的属组。

第 5 章

查找与检索

5.1 find

根据文件名查找

find [OPTION] path… [expression]

在目录中搜索文件，path指定目录路径，系统从这里开始沿着目录树向下查找文件。它是一个路径列表，相互用空格分离，如果不写path，那么默认为当前目录。Expression 是find命令接受的表达式，find命令的所有操作都是针对表达式的。

一条最常用的find命令——在当前目录及子目录下查找所有以file开头的文件名。

```
$ find . -name 'file*'

$ find / -name 'vimrc'

$ find ~ -name '*.c'
```

5.2 grep

根据内容检索

```
grep [options] PATTERN [FILE...]
```

在指定文件中搜索特定的内容，并将含有这些内容的行输出到标准输出。若不指定文件名，则从标准输入读取。

[options]部分包含的主要参数：

```
-c：只输出匹配行的计数。

-I：不区分大小写（只适用于单字符）。

-h：查询多文件时不显示文件名。
```

```
-l：查询多文件时只输出包含匹配字符的文件名。

-n：显示匹配行及行号。

-s：不显示不存在或无匹配文本的错误信息。

-v：显示不包含匹配文本的所有行。

-R: 连同子目录中所有文件一起查找。
```

比如到系统头文件目录下查找所有包含printf的文件

```
$ grep 'printf' /usr/include -R
```

第 6 章

安装卸载软件

6.1 apt-get

更新源服务器列表

```
sudo vi /etc/apt/sources.list
```

更新完服务器列表后需要更新下源

```
sudo apt-get update  更新源
sudo apt-get install package 安装包
sudo apt-get remove package 删除包
sudo apt-cache search package 搜索软件包
sudo apt-cache show package  获取包的相关信息，如说明、大小、版本等
sudo apt-get install package --reinstall   重新安装包
sudo apt-get -f install   修复安装
sudo apt-get remove package --purge 删除包，包括配置文件等
sudo apt-get build-dep package 安装相关的编译环境
sudo apt-get upgrade 更新已安装的包
sudo apt-get dist-upgrade 升级系统
sudo apt-cache depends package 了解使用该包依赖那些包
sudo apt-cache rdepends package 查看该包被哪些包依赖
sudo apt-get source package  下载该包的源代码
```

```
sudo apt-get clean && sudo apt-get autoclean 清理无用的包

sudo apt-get check 检查是否有损坏的依赖
```

6.2 deb包安装

```
安装deb软件包命令： sudo dpkg -i xxx.deb

删除软件包命令：  sudo dpkg -r xxx.deb

连同配置文件一起删除命令： sudo dpkg -r --purge xxx.deb

查看软件包信息命令： sudo dpkg -info xxx.deb

查看文件拷贝详情命令： sudo dpkg -L xxx.deb

查看系统中已安装软件包信息命令： sudo dpkg -l

重新配置软件包命令：  sudo dpkg-reconfigure xxx
```

6.3 原码安装

```
1. 解压缩源代码包
2. cd dir
3. ./configure
检测文件是否缺失，创建Makefile,检测编译环境
4. make
编译源码，生成库和可执行程序
5. sudo make install
把库和可执行程序，安装到系统路径下
```

第 7 章

磁盘管理

7.1 mount

命令格式：

mount [-t vfstype] -o options device dir

其中：

* -t vfstype 指定文件系统的类型，通常不必指定。mount 会自动选择正确的类型。常用类型有：

光盘或光盘镜像：iso9660

DOS fat16文件系统：msdos

Windows 9x fat32文件系统：vfat

Windows NT ntfs文件系统：ntfs

Mount Windows文件网络共享：smbfs

UNIX(LINUX) 文件网络共享：nfs

* -o options 主要用来描述设备或档案的挂接方式。常用的参数有：

loop：用来把一个文件当成硬盘分区挂接上系统

ro：采用只读方式挂接设备

rw：采用读写方式挂接设备

iocharset：指定访问文件系统所用字符集

* device 要挂接(mount)的设备。
* dir设备在系统上的挂接点(mount point)。

7.1.1 挂接光盘镜像文件

由于近年来磁盘技术的巨大进步，新的电脑系统都配备了大容量的磁盘系统，在Windows下许多人都习惯把软件和资料做成光盘镜像文件通过虚拟 光驱来使用。这样做有许多好处：一、减轻了光驱的磨损;二、现在硬盘容量巨大存放几十个光盘镜像文件不成问题，随用随调十分方便;三、硬盘的读取速度要远 远高于光盘的读取速度，CPU占用率大大降低。其实linux系统下制作和使用光盘镜像比Windows系统更方便，不必借用任何第三方软件包。

1.从光盘制作光盘镜像文件。将光盘放入光驱，执行下面的命令。

```
#cp /dev/cdrom /home/sunky/mydisk.iso 或
#dd if=/dev/cdrom of=/home/sunky/mydisk.iso
```

注：执行上面的任何一条命令都可将当前光驱里的光盘制作成光盘镜像文件/home/sunky/mydisk.iso

2.文件和目录制作成光盘镜像文件，执行下面的命令。

```
#mkisofs -r -J -V mydisk -o /home/sunky/mydisk.iso /home/sunky/ mydir
```

注：这条命令将/home/sunky/mydir目录下所有的目录和文件制作成光盘镜像文件/home/sunky/mydisk.iso，光盘卷标为：mydisk

3.光盘镜像文件的挂接(mount)

```
#mkdir /mnt/vcdrom
```

注：建立一个目录用来作挂接点(mount point)

```
#mount -o loop -t iso9660 /home/sunky/mydisk.iso /mnt/vcdrom
```

注：使用/mnt/vcdrom就可以访问盘镜像文件mydisk.iso里的所有文件了。

7.1.2　挂载U盘

1.检测存储设备名称

```
sudo fdisk -l
```

2.挂载存储设备sdb1到挂载点/mnt目录

```
sudo mount /dev/sdb1 /mnt
```

3.访问/mnt

4.卸载/mnt

```
sudo umount /mnt
```

7.2 umount

卸载命令

```
sudo umount 挂在点
```

7.3 dd

dd:拷贝

例1：拷贝光碟(注意，你的光碟是标准的 iso9660格式才可以这么做唷！)

```
dd if=/dev/cdrom of=cdrom.iso
```

例2：将文件sfile拷贝到文件 dfile中。

```
$ dd if=sfile of=dfile
```

例3：创建一个100M的空文件

```
dd if=/dev/zero of=hello.txt bs=100M count=1
```

/dev/null，外号叫无底洞，你可以向它输出任何数据，它通吃，并且不会撑着！

/dev/zero,是一个输入设备，你可你用它来初始化文件,从里面读出来的数据都是0。

第 8 章

压缩包管理

8.1 tar

tar [主选项+辅选项] 文件或者目录

tar可以为文件和目录创建档案。利用tar命令用户可以为某一特定文件创建档案（备份文件），也可以在档案中改变文件，或者向档案中加入新的文件。使用该命令时，主选项是必须要有的，辅选项是辅助使用的，可以选用。

主选项包括：

```
c 创建新的档案文件。如果用户想备份一个目录或是一些文件，就要选择这个选项。

r 把要存档的文件追加到档案文件的未尾。

t 列出档案文件的内容，查看已经备份了哪些文件。

u 更新文件。用新增的文件取代原备份文件，如果在备份文件中找不到要更新的文件，则把它追加到备份文件的最后。

x 从档案文件中释放文件。（常用）
```

辅选项包括：

```
f 使用档案文件或设备，这个选项通常是必选的。（常用）

k 保存已经存在的文件。

m 在还原文件时，把所有文件的修改时间设定为现在。

M 创建多卷的档案文件，以便在几个磁盘中存放。

v 详细报告tar处理的文件信息。如无此选项，tar不报告文件信息。（常用）

w 每一步都要求确认。

z 用gzip来压缩/解压缩文件，加上该选项后可以将档案文件进行压缩，但还原时也一定要使用该选项进行解压缩。（常用）
```

```
j 用bzip2来压缩/解压缩文件，加上该选项后可以将档案文件进行压缩，但还原时也一定要使用该选项进行解压缩。（常用）
```

要将文件备份到一个特定的设备，只需把设备名作为备份文件名。

打包：

```
tar cvf dir.tar dir

tar xvf dir.tar dir
```

打gz压缩包：

```
tar zcvf dir.tar.gz dir

tar zxvf dir.tar.gz
```

打bz2压缩包:

```
tar jcvf dir.tar.bz2 dir

tar jxvf dir.tar.bz2
```

指定目录解压缩：

```
tar zxvf dir.tar.gz -C ~/test
```

8.2 rar

打包：把dir压缩成newdir.rar

```
rar a -r newdir dir
```

解包：把newdir.rar解压缩到当前目录

```
unrar x newdir.rar
```

8.3 zip

打包：

```
zip -r dir.zip dir
```

解包：

```
unzip dir.zip
```

第 9 章

进程管理

9.1 who

查看当前在线上的用户情况。所有的选项都是可选的，不使用任何选项时，who命令将显示以下三项内容：

login name：登录用户名；

terminal line：使用终端设备；

login time：登录到系统的时间。

```
xingwenpeng@ubuntu:~/demo$ who -uH
名称         线路         时间              空闲        进程号 备注
xingwenpeng tty2         2014-08-14 13:31   .          6798
xingwenpeng tty7         2014-08-14 01:31  旧         2423
xingwenpeng pts/1        2014-08-14 01:31 12:00       2843 (:0)
xingwenpeng pts/3        2014-08-14 10:39   .          2843 (:0)
```

9.2 ps

```
ps [选项]
```

ps命令用于监控后台进程的工作情况，因为后台进程是不和屏幕键盘这些标准输入/输出设备进行通信的，所以如果需要检测其情况，便可以使用ps命令了。选项部分如下：

```
-e 显示所有进程。

-f 全格式。

-h 不显示标题。

-l 长格式。
```

```
-w 宽输出。

a 显示终端上的所有进程，包括其他用户的进程。

r 只显示正在运行的进程。

x 显示没有控制终端的进程。
```

这个命令参数有很多，但一般的用户只需掌握一些最常用的命令参数就可以了。 最常用的三个参数是u、a、x， 我们首先以root身份登录系统，查看当前进程状况

```
xingwenpeng@ubuntu:~$ ps aux
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   3672  2008 ?        Ss   08:46   0:01 /sbin/init

xingwenpeng@ubuntu:~$ ps ajx
 PPID   PID  PGID   SID TTY      TPGID STAT   UID   TIME COMMAND
 4592  6948  6948  4592 pts/3     6948 R+    1000   0:00 ps ajx

xingwenpeng@ubuntu:~$ ps -Lf 2423
UID        PID  PPID   LWP  C NLWP STIME TTY      STAT   TIME CMD
1000      2423  2282  2423  0    4 08:46 ?        Ssl    0:00 gnome-session --session=ubuntu
1000      2423  2282  2465  0    4 08:46 ?        Ssl    0:00 gnome-session --session=ubuntu
1000      2423  2282  2466  0    4 08:46 ?        Ssl    0:00 gnome-session --session=ubuntu
1000      2423  2282  2468  0    4 08:46 ?        Ssl    0:00 gnome-session --session=ubuntu
```

Head标头：

```
USER    用户名
UID     用户ID（User ID）
PID     进程ID（Process ID）
PPID    父进程的进程ID（Parent Process id）
SID     会话ID（Session id）
%CPU    进程的cpu占用率
%MEM    进程的内存占用率
VSZ     进程所使用的虚存的大小（Virtual Size）
RSS     进程使用的驻留集大小或者是实际内存的大小，Kbytes字节。
TTY     与进程关联的终端（tty）
STAT    进程的状态：进程状态使用字符表示的（STAT的状态码）
R 运行    Runnable (on run queue)            正在运行或在运行队列中等待。
S 睡眠    Sleeping                休眠中，受阻，在等待某个条件的形成或接受到信号。
I 空闲    Idle
Z 僵死    Zombie (a defunct process)        进程已终止，但进程描述符存在，直到父进程调用wait4()系统调用后释放。
D 不可中断    Uninterruptible sleep (ususally IO)    收到信号不唤醒和不可运行，进程必须等待直到有中断发生。
T 停止    Terminate                进程收到SIGSTOP，SIGSTP，SIGTIN，SIGTOU信号后停止运行运行。
P 等待交换页
W 无驻留页    has no resident pages        没有足够的记忆体分页可分配。
X 死掉的进程
```

```
< 高优先级进程                   高优先序的进程
N 低优先     级进程                   低优先序的进程
L 内存锁页     Lock                有记忆体分页分配并缩在记忆体内
s 进程的领导者（在它之下有子进程）；
l 多进程的（使用 CLONE_THREAD, 类似 NPTL pthreads）
+ 位于后台的进程组
START     进程启动时间和日期
TIME     进程使用的总cpu时间
COMMAND     正在执行的命令行命令
NI     优先级(Nice)
PRI     进程优先级编号(Priority)
WCHAN     进程正在睡眠的内核函数名称；该函数的名称是从/root/system.map文件中获得的。
FLAGS     与进程相关的数字标识
```

9.3 jobs

用来显示当前shell 下正在运行哪些作业（即后台作业）。

```
$ cat
```

（按下Ctrl-z挂起当前进程）

```
[1]+  Stopped                 cat

$ cat
```

（按下Ctrl-z挂起当前进程）

```
[1]+  Stopped                 cat

$ jobs

[1]-  Stopped                 cat

[2]+  Stopped                 cat
```

第一列方括号中的数字表示作业序号，它是由当前运行的shell 分配的，而不是由操作系统统一分配的。在当前shell 环境下，第一后台作业的作业号为1，第二作业的作业号为2，等等。第二列中的“+”号表示相应作业的优先级比“-”号对应作业的优先级高。第三列表明作业状态，是否为运行、中断、等待输入或停止等。最后列出的是创建当前这个作业所对应的命令行。

9.4 fg

```
fg [job…]
```

把指定的后台作业或挂起作业移到前台运行。 参数job是一个或多个进程的PID，或者是命令名称，或者是作业号（作业号前面要带一个%号）。

通常在shell中输入命令启动进程后，如果该进程需要与用户交互，那么此后用户的键盘输入都被该进程读取，直到该进程退出后才出现shell提示符$，这种进程为前台进程。

如果在命令行的末尾加上&字符，则shell为这个命令创建一个后台进程，它虽然也可以输出到屏幕，但是不能读取键盘输入，不管执行命令的进程有没有退出都立刻回到shell提示符接受下一条命令的输入。如果该进程也需要读取键盘输入，则被挂起等待直到用户用fg命令把它变成前台进程。如果一个命令需要较长的处理时间并且不需要与用户交互，就适合把它放在后台执行。

9.5 bg

```
bg [job…]
```

把被挂起的进程提到后台执行。 其中，job是一个或多个进程的PID、命令名称或者作业号，在参数前要带%号。

```
$ cat
```

（按下Ctrl-z挂起当前进程）

```
[1]+  Stopped                 cat
$ bg %1
[1]+ cat &
$（再回车一次）
[1]+  Stopped                 cat
$ fg %1
cat
（按Ctrl-d输入文件结束符）
$
```

9.6 kill

向指定进程发送信号

```
kill [ -signal | -s signal ] pid ...
```

查看信号编号

```
kill -l  [ signal ]
```

给一个进程发信号，或终止一个进程的运行。

```
$ cat
（按Ctrl-z挂起当前进程）
[1]+  Stopped                 cat
$ ps
      PID TTY          TIME CMD
     5819 pts/1    00:00:00 bash
     5893 pts/1    00:00:00 cat
 5894 pts/1    00:00:00 ps
$ kill -SIGKILL 5893
$（再次按回车键）
[1]+  Killed                  cat
$
```

kill命令如果不带参数而直接跟pid，就是发给该进程SIGTERM信号，大部分进程收到该信号就会终止。但是被挂起的进程不能处理信号，所以必须发SIGKILL信号，由系统强制终止进程。

9.7 env

查看当前进程环境变量

```
$env
```

* vim ~/.bashrc

配置当前用户环境变量

* vim /etc/profile

配置系统环境变量,配置时需要有root权限

第 10 章

用户管理

10.1 创建用户

```
sudo useradd -s /bin/bash -g xingwenpeng -d /home/xingwenpeng -m xingwenpeng
sudo useradd -s /bin/sh -g group -G adm,root xwp
```

此命令新建了一个用户xwp，该用户的登录Shell是/bin/sh，他属于group用户组，同时又属于adm和root用户组，其中group用户组是其主组。

```
-s 指定新用户登陆时shell类型

-g 指定所属组，该组必须已经存在

-G 指定附属组，该组必须已经存在

-d 用户家目录

-m 用户家目录不存在时，自动创建该目录
```

10.2 设置用户组

```
sudo groupadd xingwenpeng
```

10.3 设置密码

```
sudo passwd xingwenpeng
```

10.4 切换用户

su 用户名

```
su xingwenpeng
```

10.5 root用户

变成root用户

```
sudo su
```

设置root密码

```
passwd
```

10.6 删除用户

userdel 选项 用户名
常用的选项是-r，他的作用是把用户的主目录一起删除。 例如：

```
sudo userdel -r xingwenpeng
```

此命令删除用户xingwenpeng在系统文件（主要是/etc/passwd，/etc/shadow，/etc/group等）中的记录，同时删除用户的主目录。

第 11 章

网络管理

11.1 ifconfig

1.查看网卡信息

ifconfig

2.关闭网卡

sudo ifconfig eth0 down

3.开启网卡eth0

sudo ifconfig eth0 up

4.给eth0配置临时IP

sudo ifconfig eth0 IP

11.2 ping

ping [选项] 主机名/IP地址

查看网络上的主机是否在工作。它向该主机发送ICMP ECHO_REQUEST包。有时我们想从网络上的某台主机上下载文件，可是又不知道那台主机是否开着，就需要使用ping命令查看。

命令中各选项的含义如下：

```
-c 数目 在发送指定数目的包后停止。
-d 设定SO_DEBUG的选项。
-f 大量且快速地送网络封包给一台机器，看它的回应。
-I 秒数 设定间隔几秒送一个网络封包给一台机器，预设值是一秒送一次。
-l 次数 在指定次数内，以最快的方式送封包数据到指定机器（只有超级用户可以使用此选项）。
-q 不显示任何传送封包的信息，只显示最后的结果。
-r 不经由网关而直接送封包到一台机器，通常是查看本机的网络接口是否有问题。
-s 字节数 指定发送的数据字节数，预设值是56，加上8字节的ICMP头，一共是64ICMP数据字节。
```

11.3 netstat

```
netstat [选项]
```

显示网络连接、路由表和网络接口信息，可以让用户得知目前都有哪些网络连接正在运作。命令中各选项的含义如下：

```
-a 显示所有socket，包括正在监听的。
-c 每隔1秒就重新显示一遍，直到用户中断它。
-i 显示所有网络接口的信息，格式同“ifconfig -e”。
-n 以网络IP地址代替名称，显示出网络连接情形。
-r 显示核心路由表，格式同“route -e”。
-t 显示TCP协议的连接情况。
-u 显示UDP协议的连接情况。
-v 显示正在进行的工作。
```

11.4 nslookup

```
nslookup name
```

查询一台机器的IP地址和其对应的域名。它通常需要一台域名服务器来提供域名服务。如果用户已经设置好域名服务器，就可以用这个命令查看不同主机的IP地址对应的域名。

不带参数使用nslookup命令时，出现提示符“>”，在后面输入要查询的IP地址或域名并回车即可。如果要退出该命令，输入exit并回车即可。

```
xingwenpeng@ubuntu:~$ nslookup
> www.itcast.cn
Server:		127.0.0.1
Address:	127.0.0.1#53

Non-authoritative answer:
Name:	www.itcast.cn
Address: 115.29.149.42
>
```

11.5 finger

finger [-lmsp] user [user@host …] 查询用户的信息，通常会显示系统中某个用户的用户名、主目录、停滞时间、登录时间、登录shell等信息。如果要查询远程机上的用户信息，需要在用户名后面接“@主机名”，采用[用户名@主机名]的格式，不过要查询的网络主机需要运行finger守护进程。 命令中各选项的含义如下：

-s 显示用户的注册名、实际姓名、终端名称、写状态、停滞时间、登录时间等信息。

-l 除了用-s选项显示的信息外，还显示用户主目录、登录shell、邮件状态等信息，以及用户主目录下的.plan、.project和.forward文件的内容。

-p 除了不显示.plan文件和.project文件以外，与-l选项相同。

```
xingwenpeng@ubuntu:~$ finger xingwenpeng
Login: xingwenpeng                          Name: xingwenpeng
Directory: /home/xingwenpeng                Shell: /bin/bash
On since Mon Sep  8 08:55 (CST) on tty7    14 hours 48 minutes idle
On since Mon Sep  8 21:57 (CST) on pts/1 from :0
   11 minutes 18 seconds idle
On since Mon Sep  8 23:12 (CST) on pts/2 from :0
   6 seconds idle
No mail.
No Plan.
```

第 12 章

常用服务器构建

12.1 ftp

12.1.1 ftp服务器

1.安装vsftpd服务器

```
sudo apt-get install vsftpd
```

2.配置vsftpd.conf文件

```
sudo vi /etc/vsftpd.conf
```

添加下面设置

```
anonymous_enable=YES
anon_root=/home/xinwenpeng/ftp
no_anon_password=YES
write_enable=YES
anon_upload_enable=YES
anon_mkdir_write_enable=YES
```

3.重启服务器，重新加载/etc/vsftpd.conf配置文件

```
sudo /etc/init.d/vsftpd restart
```

4.进入你的/home/xingwenpeng/ftp目录下创建一个空目录，供用户上传

```
cd ~/ftp

mkdir anonymous

chmod 777 anonymous
```

5.测试上传功能，登陆ftp服务器，进入到anonymous目录

```
ftp  IP

cd anonymous
```

6.上传命令，可以把你当前目录下的文件上传到ftp服务器的anonymous目录

```
put  somefile
```

12.1.2 ftp客户端

Ubuntu 默认已经安装ftp客户端

12.1.3 lftp客户端

lftp也是一种ftp客户程序。它是以文本方式操作的，但是比起图形界面更为方便。lftp几乎具有bash的所有方便功能，Tab 补全，bookmark， queue， 后台下载等可以得到支持。用法与ftp类似，主要的指令如下：

```
put 上传文件

mput 上传多个文件

get 下载文件

mget 下载多个文件

mirror 下载整个目录及其子目录

mirror -R 上传整个目录及其子目录

!command 调用本地shell执行命令command
```

注意，有的发行版可能缺省没有安装lftp工具，需要用户自己安装。如果是Debian或Ubuntu系统，则安装lftp软件包。

```
sudo apt-get install lftp
```

12.2 nfs

1.安装nfs服务器

```
sudo apt-get install nfs-kernel-server
```

2.设置/etc/exports配置文件

```
sudo vi /etc/exports
```

添加这行配置

```
/home/用户名/nfs   *(rw,sync,no_root_squash)
```

3.在用户目录下创建nfs目录

```
mkdir /home/用户名/nfs
```

4.重启服务器，重新加载配置文件

```
sudo /etc/init.d/nfs-kernel-server restart
```

5.在/home/用户名/nfs目录下创建测试文件hello

```
cd /home/用户名/nfs

touch hello
```

6.测试服务器，把服务器共享目录nfs挂在到/mnt节点

```
sudo mount -t nfs -o nolock -o tcp IP:/home/用户名/nfs  /mnt
```

7.进入/mnt目录可以看到hello文件，表示构建成功
8.卸载网络共享目录

```
sudo umount /mnt
```

12.3 ssh

1.安装ssh服务器

```
sudo apt-get install openssh-server
```

2.远程登陆

```
ssh 用户名@IP
```

12.4 telnet

明文传输数据，请大家自行测试

第 13 章

其它命令

13.1 终端翻页

```
Shift-pageup
Shift-pagedown
```

13.2 man

看手册(叫做manual或man page)。每一个命令和系统函数都有自己的man page。

```
man man
man read  查看read命令的man page
man 2 read  查看read系统函数的man page(在第二个section中，表示为read(2))
man -k read  以read为关键字查找相关的man page
```

13.3 clear

清屏。使光标和提示符回到屏幕第一行。

```
快捷键：Ctrl-l
```

13.4 alias

alias [-p] name=value …

将value字符串起个别名叫name，以后在命令行输入name，shell自动将其解释为value，如果不带参数执行本命令，或以参数-p执行，则显示当前定义的别名列表。

```
$ alias
alias ls='ls --color=auto'
alias rm='rm -i'
```

13.5 echo

```
echo [-n] 字符串
```

在显示器上显示一段文字，一般起到一个提示的作用。其中选项n表示输出文字后不换行；字符串可以加引号，也可以不加引号。用echo命令输出加引号的字符串时，将字符串原样输出；用echo命令输出不加引号的字符串时，将字符串中的各个单词作为字符串输出，各字符串之间用一个空格分割。

查看上一个程序退出数值，正常情况程序退出值是0

```
echo $?
```

13.6 date

查看当前时间

13.7 umask

```
umask [-p] -S [mode]
```

umask指定用户创建文件时的掩码，其中的mode和chmod的命令中的格式一样。如果不用mode参数，则显示当前的umask设置。如果用-S参数，则以符号形式显示设置。

```
$ umask
0022
$ umask -S
u=rwx,g=rx,o=rx
```

比如该用户touch或gedit创建一个文件，则其默认权限为-rw-r-r—，如果该用户创建一个可执行文件(比如编译生成的程序)，则其默认权限为-rwxr-xr-x。也就是说，由于umask的设定，创建的文件默认是不具有g的w权限和o的w权限的，除非用chmod更改权限。

第 14 章

关机重启

关机重启这些操作都需要有root权限

14.1 poweroff

14.2 shutdown

shutdown -t 秒数 [-rkhncfF] 时间 [警告讯息]

```
-t 秒数 : 设定在切换至不同的runlevel之前，警告和删除二讯号之间的延迟时间(秒).

-k : 仅送出警告讯息文字，但不是真的要 shutdown.

-r : shutdown 之後重新开机.

-h : shutdown 之後关机.

-n : 不经过 init , 由 shutdown 指令本身来做关机动作.(不建议你用)

-f : 重新开机时，跳过 fsck 指令，不检查档案系统.

-F : 重新开机时，强迫做 fsck 检查.

-c : 将已经正在 shutdown 的动作取消.
```

例子：

```
shutdown -r now 立刻重新开机
shutdown -h now 立刻关机
shutdown -k now 'Hey! Go away! now....' 发出警告讯息，但没有真的关机
shutdown -t3 -r now 立刻重新开机，但在警告和删除processes 之间，延迟3秒钟.
shutdown -h 10:42 'Hey! Go away!' 10:42 分关机
shutdown -r 10 'Hey! Go away!' 10 分钟後关机
shutdown -c 将刚才下的 shutdown 指令取消,必须切换至其它tty，登入之後，才能下此一指令.
shutdown now 切换至单人操作模式(不加任何选项时)
```

注意事项：
时间参数务必要加：不是用 now，便是用 hh:mm 或 mm
now 其实就是 0 的意思.

14.3 reboot

14.4 查看内核版本信息

```
uname -a
```

14.5 查看发行版信息

```
lsb_release -a
```

14.6 查看空闲内存

```
free -m
```

第 15 章

练习

1.创建test目录，在里面创建aa bb cc三个目录，在aa里创建hello文件，在bb里创建world目录，在cc里创建itcast.c,然后执行tree/ls -R,最后删除test

第 16 章

需要安装的组件

```
sudo apt-get install openssh-server

sudo apt-get install nfs-kernel-server

sudo apt-get install vsftpd
```